Execution Plan
SQL Server 2017

Thomas LeBlanc
Data Warehouse Architect
About me...

- SQL Server Architect (MVP)
  - DW BI Architect
  - Still do DBA Work

- Database Normalization “Nut”
  - “Crazy” about Dimensional Modeling
  - Retired Developer

- TheSmilingDBA (twitter, gmail & blog)

- PASS Volunteer
  - Past VC Chair – Data Arch and Excel BI
  - Nomination Committee 2016, 2017
  - Baton Rouge SQL Server User Group
  - Baton Rouge Analytics User Group
Assumptions about you...

- Write T-SQL
- Limited or Not Used Execution Plans
- Developer or DBA or cool IT Dude
- Sense of Humor
What are we talking about?

- Terminology
- History
- Demo, Demo, Demo,
- New Features
History

- Text - Enterprise manager (no SSMS)
- 2005 - Graphical Plans
- 2008 – Missing Index
- 2012 – Warnings/Memory Grant
- 2014 – Operator Cost
- 2016 – Query Store
  - Compare Plans
  - Live Query Statistics!!!
- 2017
  - Improved Query Store
  - Adapted Joins!!!
- SSMS – Analyze Actual Query Plan
SET SHOWPLAN_TEXT ON
GO
-- Select all columns
FROM [dbo].[DatabaseLog]
GO

(1 row affected)

SalesOrderHeader

FROM [Sales].[SalesOrderHeader]
WHERE SubTotal < 300

(1 row affected)

SalesOrderHeader

-- Filters WHERE: ([AdvWrk16].[Sales].[SalesOrderHeader].[SubTotal] < (4100.0000))
|-- Compute Scalar DEFINE: ([AdvWrk16].[Sales].[SalesOrderHeader].[TotalDue] = [AdvWrk16].[Sales].[SalesOrderHeader].[TotalDue])
|-- Compute Scalar DEFINE: ([AdvWrk16].[Sales].[SalesOrderHeader].[TotalDue] = isnull([AdvWrk16].[Sales].[SalesOrderHeader].[TotalDue])
|-- Clustered Index Scan OBJECT: ([AdvWrk16].[Sales].[SalesOrderHeader].[PK_SalesOrderHeader_SalesOrderID])

(4 rows affected)
Terminology

- Query Plan
  - Use by Optimizer
  - Optimizer Articles – SQLServerCentral.com (Gail Shaw)
  - Estimated versus Actual
    - Now Live Query Stats
- Scans
  - Heap for Tables
  - Clustered Index Scan
  - Non-clustered Index Scan
- Seek
  - Good use of an index
More Terminology

- Lookup
  - After Seek, more data is extracted
  - Key versus RID Lookup
    - Key – table has clustered index
    - RID – Heap with no clustered index
Demo

- Scan, Seeks and more
And More Terminology

- **Loops**
  - **Nested**
    - Each row on outer...
    - Joined with inner rows
  - **Merge Join**
    - Like a Zipper
    - Out and In rows are sorted
    - Same Order
  - **Hash**
    - Complicated
    - Not bad, but most expensive
    - Join columns are Hashed
      - Data put is ~4 buckets
      - Then data sets are hashed and joined
Demo

- Loops

```sql
SELECT [TSQL] FROM [dbo].[DatabaseLog] WHERE [Data

Nested Loops
(Inner Join)
Cost: 0%

Index Seek (NonClustered)
[DatabaseLog].[idxDatabaseLog]
Cost: 25%

Key Lookup (Clustered)
[DatabaseLog].[PK_DatabaseLog]
Cost: 75%
```
Other operators

- Compute Scaler
  - Concatenate Last and First Name
- Sort
  - ORDER BY, DISTINCT or GROUP BY
- Aggregate
  - GROUP BY and COUNT/SUM/etc.
- Properties (more information)
- Insert and Updates (Deletes)
Demo

- Other Operators
Parallelism & Cost Above Threshold

- Using Multiple Processors (threads)
- MAXDOP configuration
  - Set to 0 by Default
- Cost Threshold
  - Defaults to 5
  - Reduce to get more parallelism
  - Increase to get less parallelism
- NEW!!! 2017
  - Database Level
  - Gotcha: Connection in DB
Demo

- MAXDOP & Live Query Stats
Query Store

- Store Plans in DB
  - Has to be turned on
- Dashboard
  - Regressed Queries
  - Top Consumers
  - High Variation
  - Overall Resource Consumption
  - Tracked Queries
  - Queries with Forced Plans
Demo

- Query Store
- Analyze QP
Thanks!!!

- Thomas LeBlanc
- Twitter - @TheSmilingDBA
- Thomas-LeBlanc.com
- TheSmilingDBA@gmail.com